An Implementation of Zero Trust Architecture by p≡p

Whitepaper for cyber security experts and policy makers

by Volker Birk, Hartmut Goebel et al.

p≡p security, Aeschstrasse 4 8834 Schindellegi, Switzerland

1 Executive Summary

This white paper outlines a unique solution for Zero Trust Architecture (ZTA). The software suite developed by p≡p is the first commercial software suite designed and programmed to fully meet the requirements set out by the National Institute of Standards and Technology (NIST)[1]. The p≡p software suite fulfils the requirements for the highest possible standard of CISA's Zero Trust Maturity Model (cf. https://zerotrust.cyber.gov).

p≡p is a comprehensive software solution needed to achieve full ZTA. p≡p allows to implement ZTA across your entire IT infrastructure with the highest possible granularity.

ZTA can be achieved within one software upgrade cycle. Installation is on the endpoints; these can be servers, virtual machines, devices etc. or single applications; it can be rolled out gradually within existing IT infrastructure, seamlessly connecting across any combination of on-premise / cloud, not needing a modification of any existing software application. Legacy systems can stay in place. Over time, existing security elements, such as TLS can be decommissioned.

p≡p's full scale software suite offers solutions for the core components of ZTA (i.e. policy engine and policy enforcement point) as well as for all of the functional components as defined by NIST (see chart I). The only functional component on top of p≡p is endpoint security, and there are excellent solutions available which are fully compatible with p≡p.

p≡p solutions provide:

- fully automated identity & access management for all users and resources in the cloud, on premise and remote.
- secure all data in transit end-to-end with peer-to-peer encryption without a central element.
- prevent any lateral movement of any un-authorized parties.
- monitor all traffic and activity.

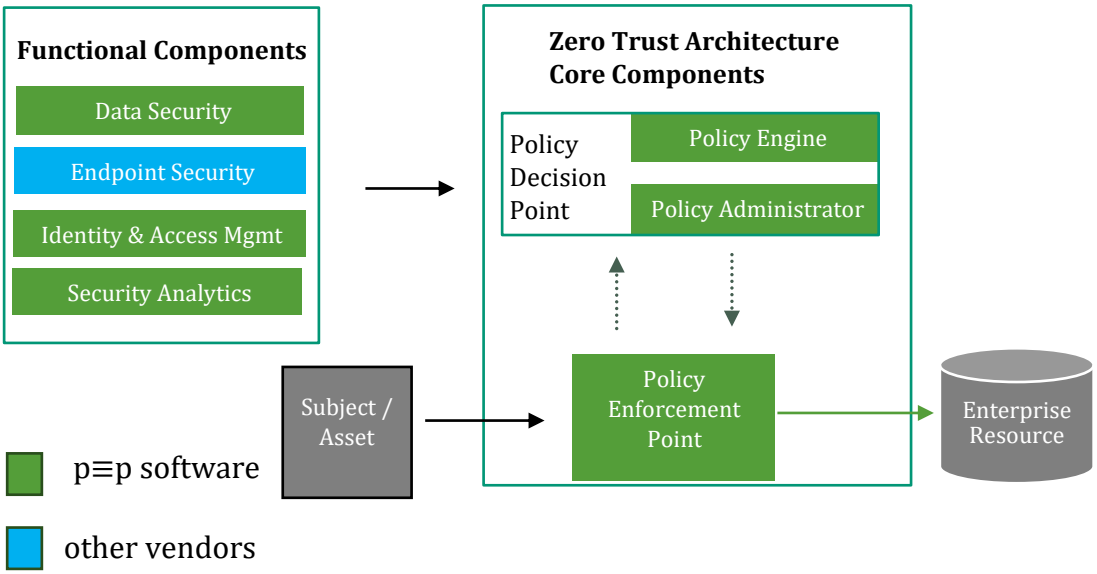


Chart I: ZTA High Level Architecture by NIST

[1] The standard is set out in NIST SP 800-207

The cornerstones of the solution are:

1. Assumes Zero Trust between all systems.
2. Makes virtually all network management decentralized.
3. Secures all communication between systems by cryptographical enforcement.
4. Secures access to all resources by decentralized enforcement of policies.
5. Fully supports mixed cloud/on-premises environments.
6. Implements everything peer-to-peer without any central element.
7. Secure existing infrastructure and existing network communication.

p≡p is a peer-to-peer software which has fully automated identity, trust and key-management with no central element; the key cornerstone to achieve ZTA.

A more detailed explanation of the concepts behind the p≡p technology can be found under: https://dev.pep.foundation/Concepts.

2 Identity and Access management

Given the decentralized nature of today's IT infrastructure it is virtually impossible to manage identity and access with central systems and achieve ZTA. p≡p's answer is a fully decentralized system with peer-to-peer software and a blockchain. The peer-to-peer software provides the full functionality of a ZTA policy enforcement point while the policy engine is implemented in a blockchain.

Combining these two core components of ZTA allows to achieve identity and access management fully decentralized and ZTA. The solution is highly scalable and high frequency as the p≡p peer-to-peer software is taking on most of the workload; only configuration and policy changes are executed in the blockchain.

The p≡p peer-to-peer software is installed on each endpoint. Each server, virtual machine, device etc. and single application becomes a policy enforcement point i.e. all authentication and authorization is at the point of resource or trust zone.

A fully decentralized policy engine (explanation see chapter 6) in a blockchain ensures a seamless configuration management of all users and resources. The policy engine is distributed across all endpoints. As with any blockchain, communication is peer-to-peer and there is no central element.

As a result, each endpoint contains the relevant configuration and user data from the distributed policy engine and can grant access through the peer-to-peer policy enforcement point.

In the user model, p≡p authenticates persons, whereas in the machine-to-machine model, p≡p authenticates systems, or more precisely: applications. In both cases, the authentication is also used for authenticating subsequent network communication.

The user needs to authenticate only once. p≡p integrates into the authentication system of the operating system they're running on, respectively. On a laptop, for example, the user

authenticates with its smartcard. When accessing the company's services, p≡p will automatically authenticate the user's network session.

In detail this works as follows:

1. Valid User Accounts are replicated by Distributed Policy Engine.
2. The operating system authenticates the user (including multi-factor authentication).
3. p≡p Client or Agent starts “as the authenticated user” and accesses the keys of the user.
4. p≡p Client or Agent utilizes the user’s keys to authenticate network sessions.

In the machine-to-machine model, p≡p Client or Agent starts “as the authenticated application”.

3 p≡p secures all data in transit

p≡p secures all data in transit regardless of the network location. 'Secured' is defined as: identified, sender authenticated and end-to-end encrypted.

The technology automatically encrypts the existing network traffic: establishing new connections with Trust-On-First-Use (TOFU) and a peer-to-peer public key exchange. The private key always remains on the endpoint and is used for signing each message, which ensures sender authentication.

The software is only installed on the endpoints which need to secure the communication. Endpoints can be applications, virtual machines, devices etc. All data and keys which are needed to secure the communication are generated, stored and revoked on the endpoints only. There is no central element. The communication is secured on a peer-to-peer basis. As a result, there is no single point of trust and no “winner takes it all” attack vector.

Data is secured independently on two layers: application layer and IP layer.

On the application layer, messages are encrypted when they leave the existing application by using the public keys of the communication partners. The p≡p solution allows for flexibility. An application with p≡p installed can send encrypted messages to applications that have p≡p installed and unencrypted messages, as before, to all other applications.

This flexibility allows for a gradual roll-out within a network: starting with the most sensitive communication first. Configured to not send any messages unencrypted and to send to trusted communication partners only. Out of the box, there is support for the common message formats in the corporate and banking world: E-mail, XML, MT-FIN, ISO-20022 etc.

On the IP layer, automatic encryption is enabled via IPSec. The software performs the functions of fully automated identity-, trust- and key-management software, leading to much more fine-grained control than one could ever manage with IPSec/IKE. As a result, any TLS or similar encryption becomes obsolete.

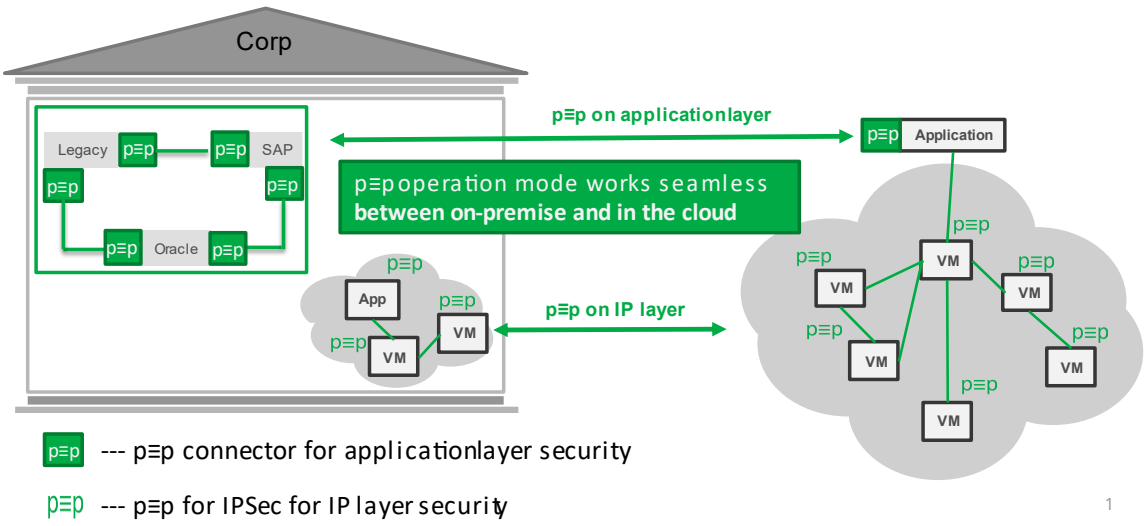


Chart II: p≡p secures all data in transit across the whole IT landscape

4 p≡p prevents lateral movement by unauthorized parties

p≡p has two modes, a “learning mode” and a “locked mode”.

In the “learning mode”, p≡p establishes new connections with Trust-On-First-Use (TOFU) and a peer-to-peer public key exchange. Thanks to machine learning, each p≡p instance learns about its communication partners. Each incoming message is checked, rated and logged, e.g., for a test bench to set up a new cloud segment. The assumption is that this is done while provisioning when a man-in-the-middle-attack on the first message exchange can be excluded.

For environments where this cannot be assumed, the approach is to use a separate verification of keys by trust words on a side channel[2]. This closes the man-In-the-middle-attack-vector entirely. The “learned data” defines legitimate communication.

In “locked mode”, no illegitimate communication can be established. If an endpoint tries to access a communication partner illegitimately, this communication will be blocked, the incident will be registered and an alert will be raised. As a result, all unauthorized communication can be prevented, which leads to blocking of all lateral movement. This is illustrated in chart III.

[2] For a detailed explanation: https://www.pep.security/docs/en/

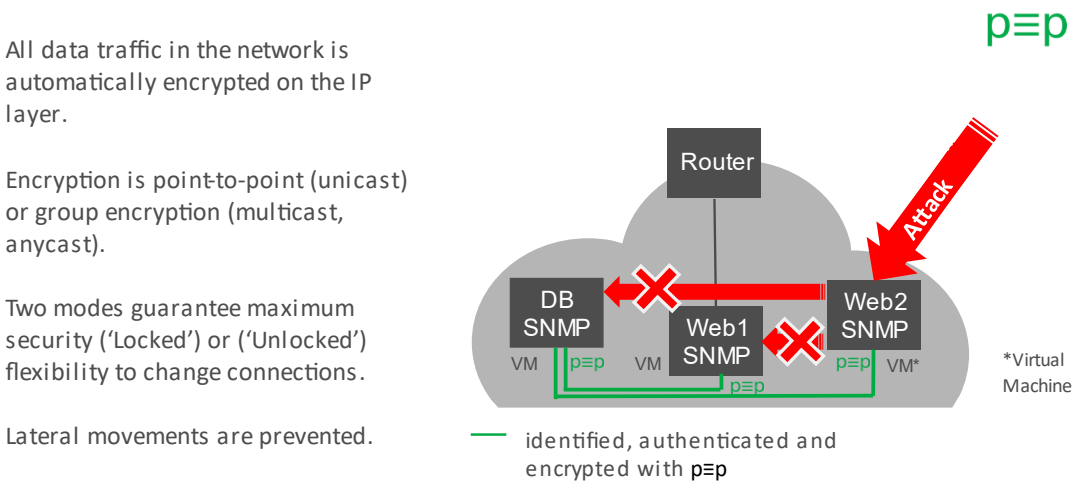


Chart III: p≡p for IPSec makes each communication endpoint a policy enforcement point

5 p≡p monitors all traffic and activity

All messages are checked, rated and logged on a local database on each endpoint. The details of the p≡p rating system can be found at: https://dev.pep.foundation/Engine/TrustRating.

Thanks to machine learning each endpoint constantly updates the knowledge about its communication partners. Suspicious messages are reported immediately.

Using the alerting function, any suspicious activity such as unauthorized communication attempts can deliver alerts to enterprise monitoring systems, allowing early detection and reaction to any attack.

6 p≡p implements the policy engine in a blockchain

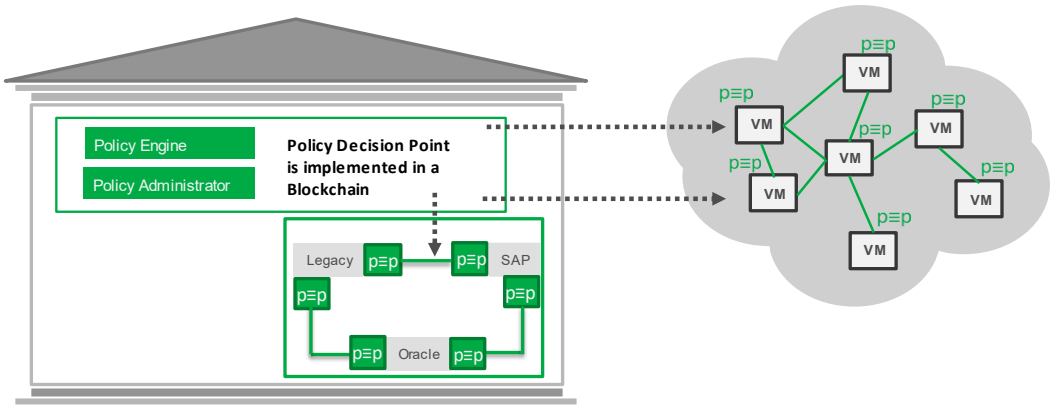
In Zero Trust Architecture as proposed by NIST, a central policy engine would be the central point of attack. It would be a "winner takes it all" instance, so additional protection is needed.

p≡p improves on the NIST standard with the use of a distributed policy engine, communicating via a private blockchain. The policy engine is distributed across all endpoints. As with any blockchain, communication is peer-to-peer and there is no central element.

The implementation of the policy engine on a blockchain has much stronger security properties than other implementation approaches. For example, a node can choose validators, which decide about the state of the blockchain, randomly.

In the p≡p concept, the distributed policy engine manages the configuration data and rules, and therefore the policy decisions. Policy enforcement is local, with decisions taken at the end-point. As a result, even in very large IT infrastructures, only a limited number of changes will have to be managed by the blockchain and consequently scaling up is not an issue.

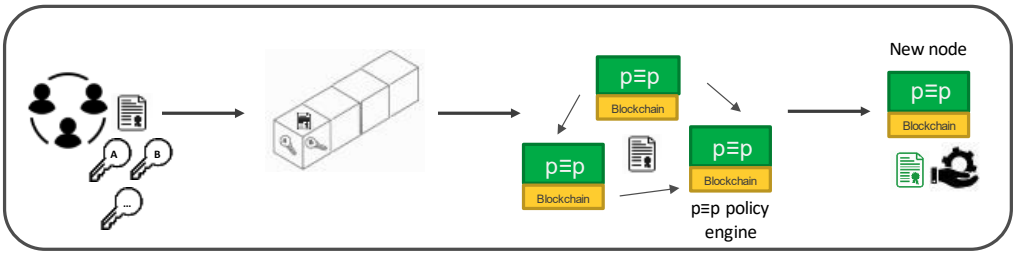
p≡p implements all policies as defined by the central Policy Engine locally and peer-to-peer.



p≡p is the Policy Enforcement Point as defined by NIST.

Chart IV: p≡p distributed policy engine implementation in a blockchain within each p≡p connector

At each end-point, there is a corresponding node of the blockchain resp. distributed policy engine, providing the relevant configuration data and rules. Configuration updates are signed off by a minimum of two people following four-eyes principle. Subsequently, configuration updates get distributed through the blockchain to all nodes. Chart V illustrates this configuration update process.



New configuration gets signed by 4 or more eye principle. An Extra Read Key is added for audit purposes.

New configuration gets added to the current chain.

New configuration chain gets distributed within peer-to-peer network.

New configuration gets deployed on the node(s) which are allowed to read the encrypted block.

Chart V: decentral configuration and audit trail with the p≡p policy engine

7 The concept of Extra Read Keys

Message inspection is a vital capability; authorized parties with legitimate mandates need to be able to “see inside” encrypted messages. The Extra Read Keys function in a similar way to “cc” on emails; the owner of the private part of the Extra Read Key can read the message, but not alter it.

Chart VI shows an application of this in the banking world: Fraud- and compliance controls need to be carried out without changing the content and sender identity of the message. Extra Read Keys can be issued to any authorized parties.

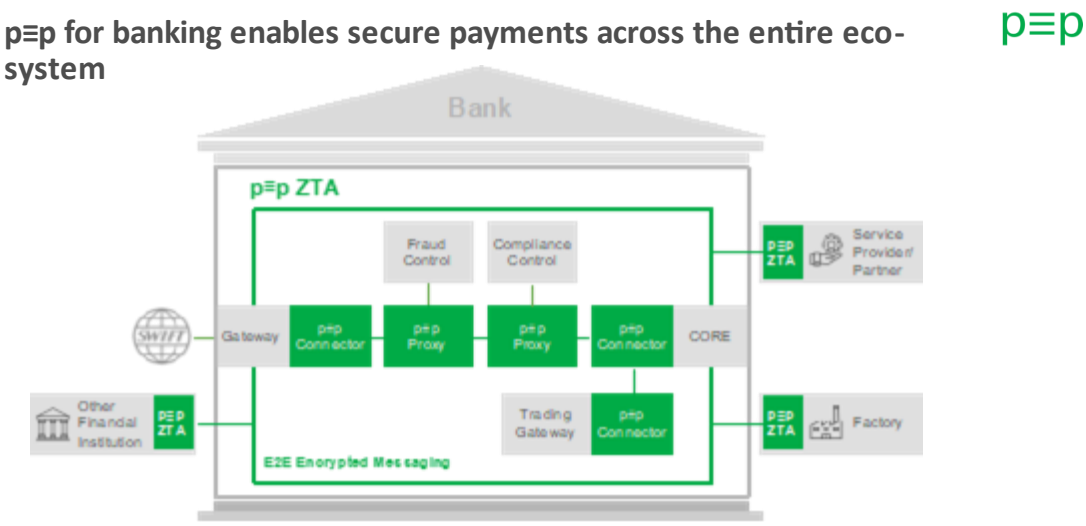


Chart VI: Extra Read Keys for authorized parties

8 Transition and implementation of ZTA

No big-bang! A roll-out is highly configurable and non-invasive; no changes to the application landscape of the existing IT infrastructure are required. The solution is non-invasive and can be rolled out gradually, starting with one application first and/or one network segment only.

The distributed policy engine can be implemented in a later stage, if no strict ZTA rules need to be applied in the beginning.

Thus, a company can start gradually on the journey to complete ZTA; protecting the most sensitive and mission critical communication and systems first. Systems which have no p≡p installation will simply continue to communicate unencrypted with the systems which do.

9 Key benefits of p≡p

p≡p offers an unparalleled security level whether it is implemented in parts or with its complete ZTA software suite. This can be best described when looking at the different attack vectors which are most relevant in any of today's cyber security use case. For any and each of those attacks, the p≡p software suite can massively reduce potential damages and significantly increase the level of protection:

Supply chain attack: The impact of supply chain attacks (i.e. Solarwinds) is massively reduced, because p≡p prevents unauthorized lateral movements and encrypts all data in transit. Attackers can only communicate from a compromised system with those other systems, which the now compromised system is allowed to communicate in the first place. This largely reduces the scope of potential attacks. With the ability to generate alerts of any suspicious activity, such as unauthorized communication attempts, the enterprise will be able to detect the attack very early and react to it.

Ransomware attack: p≡p massively reduces both the likelihood and impact of a ransomware attack. The likelihood is reduced, because p≡p deploys sender authentication. Messages which are not authorized will be put in quarantine. The impact of any potential ransomware attack is massively reduced, because unauthorized lateral movement is prevented.

Phishing and spoofing: p≡p protects against phishing and spoofing thanks to sender authentication. A man-in-the-middle-attack can be fully excluded. With p≡p for financial messages, for example, a signet will show whether the sender is trusted or not by the message recipient.

Message injection and manipulation: p≡p protects against message injection and manipulation, such as that used in the case of the attack on the Central Bank of Bangladesh, by automatically rating all incoming messages. Messages which are not trusted, will not be processed and be put into quarantine.

Social Engineering: The potential impact of social engineering in a p≡p environment is restricted because the human factor is taken out of the communication process almost completely.

Key, trust and identity management is fully automated and not dependent on manual interaction. System administration has the standard controls: 4 or 6 eyes for the approval of system updates and security policy changes. The whole system then runs decentralized, with a blockchain-based policy engine on each endpoint. While zero-day attacks are still possible, this limits their impact.

10 About p≡p

p≡p was founded with a dual structure in mind. There is a foundation and a commercial company which co-exist in a well-balanced eco-system. The commercial company launched its commercial activities focusing on enterprise customers in 2019 with the mission to provide the most secure enterprise cyber security software solutions in the market.

The mission of the foundation is to defend the inalienable human rights to **privacy** and **freedom of information** and to bring secure and trusted encryption into every application and every device. The foundation employs some of the world's best minds in the field of cryptography. It is a space where creativity to push the boundaries in cryptography and activism for privacy and freedom of information can roam freely. It owns the open-source key encryption software.

The commercial company, p≡p security, has developed a fully-fledged software suite around the key encryption software. The Swiss-based company provides **enterprise products** based on the p≡p cryptography core. p≡p security holds the exclusive commercial rights to all the source code produced by the p≡p foundation. Products are a fully-fledged software suite including p≡p for Financial Messaging, p≡p for email, p≡p for Enterprise Messaging, p≡p for IPSec and p≡p for ZTA.

More details of the concepts behind the p≡p technology can be found under: https://dev.pep.foundation/Concepts.

About the authors:

Volker Birk is a software architect and activist in the open-source space. He wrote his first software program at the age of eleven and is a renowned cyber security expert, with over 30 years of experience.

Hartmut Goebel is a security architect and developer with over 30 years of experience. As an InfoSec consultant he has been advising DAX-50 companies since 2003. He is a passionate author and speaker, holding both the CISSP and CSSLP certifications from the ISC².

Contact: info@pep.security

Websites: www.pep.security & www.pep.foundation